1. **What is OOP? List OOP concepts.**

Ans: Object-oriented programming (OOP) is a computer programming model that organizes software design around data, or objects, rather than functions and logic. An object can be defined as a data field that has unique attributes and behavior.

Oop concept—

* **Encapsulation.**This principle states that all important information is contained inside an object and only select information is exposed. The implementation and state of each object are privately held inside a defined class. Other objects do not have access to this class or the authority to make changes. They are only able to call a list of public functions or methods. This characteristic of data hiding provides greater program security and avoids unintended data corruption.
* **Abstraction.** Objects only reveal internal mechanisms that are relevant for the use of other objects, hiding any unnecessary implementation code. The derived class can have its functionality extended. This concept can help developers more easily make additional changes or additions over time.
* [**Inheritance**](https://www.techtarget.com/whatis/definition/inheritance)**.**Classes can reuse code from other classes. Relationships and subclasses between objects can be assigned, enabling developers to reuse common logic while still maintaining a unique hierarchy. This property of OOP forces a more thorough data analysis, reduces development time and ensures a higher level of accuracy.
* [**Polymorphism**](https://www.techtarget.com/whatis/definition/polymorphism)**.**Objects are designed to share behaviors and they can take on more than one form. The program will determine which meaning or usage is necessary for each execution of that object from a parent class, reducing the need to duplicate code. A child class is then created, which extends the functionality of the parent class. Polymorphism allows different types of objects to pass through the same interface.

**2. What is the difference between OOP and POP?**

1. Ans: POP is procedure-oriented programming while OOP is object-oriented programming.
2. The main focus of POP is on “**how to get the task done**” it follows the flow chart to get the task done. OOP’s main focus is on **data security** as only the objects of a class are allowed to access the attributes or function of a class.
3. The **functions** are small units of the large programs or a sub-program that execute to get the main task done. In contrast, OOP attributes and functions of the class are divided among the **objects**.
4. In POP, there is no specific accessing mode to access attributes or functions in the program. Conversely, in OOP there are three accessing modes “public”, “private”, “protected”, that are used as an accessing method to access attributes or functions.
5. POP does not support the concept of Overloading/polymorphism. On the contrary, OOP supports Overloading/Polymorphism, which means using the same function name for performing different functions. We can overload functions, constructor, and operators in OOP.
6. There is no concept of inheritance in POP whereas, OOP supports inheritance which allows using the attribute and functions of other class by inheriting it.
7. POP is less secure as compared to OOP because in OOP the access specifier limits the access to attributes or functions which increase the security.
8. In POP if some data is to be shared among all the functions in the program, it is declared globally outside all functions. While in OOP the data member of the class can be accessed through the member functions of the class.
9. In POP there is no concept of the friend function. As against, in OOP there is a concept of friend function which is not the member of the class, but because it is friend member it can access the data member and member functions of the class.
10. There is no concept of virtual classes in POP whereas in OOP, the virtual functions support polymorphism.